Undergrad Thesis

Offline E-Cash System

Authors:
Rashmi Ranjan Parida
Bineet Satapathy

Supervisor:
Prof. Sujata Mohanty

A thesis submitted in partial fulfilment of the requirements for the degree of Bachelor in Technology(B. Tech.)

in the

Computer Science Engineering
National Institute Of Technology Rourkela

National Institute Of Technology
Rourkela

May 2014

Declaration of Authorship

We, Rashmi Ranjan Parida and Bineet Satapathy, declare that this thesis titled, "Offline E-cash System" and the work presented in it are our own. We confirm that:

- This work was done wholly or mainly while in candidature for a research degree at this University.
- Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated.
- Where We have consulted the published work of others, this is always clearly attributed.
- Where We have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely our own work.
- We have acknowledged all main sources of help.
- Where the thesis is based on work done by ourself jointly with others, we have made clear exactly what was done by others and what We have contributed ourself.

Signed: Rashmi Ranjan Parida

Signed: Bineet Satapathy

Date: 10th May 2014

Certificate

This is to certify that the thesis entitled **Offline E-cash System** by **Rashmi Ranjan Parida and Bineet Satapathy** in partial fulfilment of the requirements for the award of Bachelor of Technology Degree in Computer Science and Engineering at the National Institute of Technology, Rourkela, is an authentic work carried out by them under my supervision and guidance. To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other university / institute for the award of any Degree or Diploma.

Prof. Sujata Mohanty
Dept. of Computer Science and Engineering
National Institute of Technology Rourkela
Rourkela-769008

Acknowledgements

We are very much indebted to our guide Prof. Sujata Mohanty for giving us the opportunity to work under her guidance. She motivated and inspired us through the entire duration of our work, without which this project could not have seen the light of the day.

Moreover, we convey our regards to all the other faculty members of Department of Computer Science and Engineering, NIT Rourkela for their valuable guidance and advices at appropriate times. We would also like to thank our batchmates for their help and assistance all through this project.

Last but not the least, we express our hearty gratitude to the Almighty and our parents for their blessings and support without which this task could have never been accomplished.

Rashmi Ranjan Parida
110CS0139
Dept. of Computer Science and Engineering
National Institute of Technology Rourkela

Bineet Satapathy
110CS0380
Dept. of Computer Science and Engineering
National Institute of Technology Rourkela

Abstract

The e-cash scheme and the digital content transactions are the need of the hour. In the coming years, all these digital transactions will grow tremendously. So, a secure e-cash scheme is of utmost requirement. e-cash scheme, which is untraceable and maintains the security features, make it possible for the customers and the merchants to exchange the e-cash and the merchandise with privacy. So, there is a need to design an e-cash scheme with strong cryptosystem and algorithms in order to facilitate efficient digital transactions. There are two types of e-cash systems: Offline e-cash systems and online e-cash systems. Offline e-cash systems make it possible for the customer to pay the e-coin to the merchant without any involvement of bank. In online schemes, we require the involvement of the bank. The two most fundamental security features associated with offline scheme is the anonymity and the double spending detection. The proposed scheme maintains both the above features along with unforgeability. Besides, the e-coins have their expiration date so that the bank faces no hassles and can manage its database efficiently. This feature also ensures portability as the coins can be transferred to storage devices through the networks.

Contents

List of Figures

Chapter 1

Introduction

Since the world of digitalization is first evolving, the speed of the communication and processing of data have been increasing. So in the area of payments, we need to have a strong and secure e-cash scheme that allows fast payments and can greatly reduce the time of communication. These scheme helps the customers and the merchants to exchange the e-coin and the merchandise in a secure communication channel. But despite of the advantages of the e-cash scheme, e-cash system is not being used everywhere because of non-availability of resources. Traditional cash is still persistent in many areas. One reason may be that we need to have sufficient investment on the infrastructure so that the new e-cash scheme offers considerable pros over traditional cash. So, in order to cope with this world of digitalization, we need to design an e-cash scheme with strong cryptosystem and security features embedded in it so that they can replace the traditional cash and foster faster electronic payments.
In present day most of the communication takes place on the insecure channel which makes the message vulnerable to multiple threats. On the other hand creating a secure channel is quite expensive and not scalable. While sending a message over an insecure channel such as internet to a person we must primarily provide four different security requirements such as [2, 3],

- Confidentiality: Protection our information from malicious actions.
- Integrity: Changes need to be done by only authorized entities.
- Authentication: Able to know the identity of sender.
- Non-repudiation: Either party can't deny their role later.

Cryptography only allows data confidentiality but can't allow data integrity, data authentication and Non-repudiation.

1.1 Introduction to Cryptography

Cryptography is the science of secret writing i.e. the science of transforming information to make them secure and immune attacks. Cryptography refers to the encryption and decryption of messages using symmetric-key encipherment, asymmetric-key encipherment and hashing.

1.1.1 Symmetric Key Cryptography

In this, sender will encrypt the message using encryption algorithm and receiver decrypts using decryption algorithm. Here both encryption and decryption algorithm uses same secret key. In the simplest way Alice puts the message in a box and locks the box using the shared secret key, Bob unlocks the box with the same key and takes out the message [2, 3]. Additive cipher, Affine cipher, Playfair cipher, Hill cipher etc. are the examples of Symmetric Key Cryptography.

1.1.2 Asymmetric Key Cryptography

In this, sender will encrypt the message using encryption algorithm using receiver's public key and receiver decrypts using decryption algorithm using receiver's private key. So we need two keys one is public key and other one is private key. RSA cryptosystem, Rabin cryptosystem, Elgamal cryptosystem are the examples of ASymmetric Key Cryptography.

1.1.3 Hashing

In hashing, a fixed length message digest is created out of a variable length message. Both digest and message is sent to receiver. It provides data integrity [20].

1.2 Digital Signature

A conventional Signature is included in the document. When we sign a document digitally, we send the signature as a separate document. The sender sends two documents: the message and the signature. The receipient receives both documents and verifies that the signature belongs to the supposed sender. The sender uses a signing algorithm with his private key and the receiver uses a verifying algorithm with sender's public key.

It provides message authentication, message integrity, nonrepudiation and confidentiality.Digital signature was first proposed by Diffie and Hellman[4].

1.3 Time-Stamp

Time-stamping is a technique for providing proof of existence of certain digital document or data prior to a specific time [5]. Time-stamping is now widely recognized as an important mechanism used to ensure the integrity of digital data. Time-stamping is highly required in many domains like patent submissions, electronic votes or electronic commerce. Time-stamping is usually enforced to ensure non-repudiation. A digital signature can only be legally binding if it was made when the user's certificate was still valid, and a time-stamp on a signature can successfully prove this

1.4 Classification of e-cash System

e-cash system can be classified as offline or online [1].

1.4.1 Offline e-cash

In offline e-cash system customer pays merchant with out help of bank means there is no role of bank during transaction but during deposit merchant will deposit the e-coin in the bank.

1.4.2 Online e-cash

When customer pays merchant, bank should be online means presence of bank is needed during payment. In online e-cash system all participants should be online.

1.5 Basic operations of e-cash

e-cash system contains four basic operations such as:

1.5.1 Key Generation

This includes the key generation by the Bank. Bank will generate the public keys and private keys. They are used in cryptography and Digital signature. Bank will publish the public key publicly such that all the customers and merchants can use those keys. Bank wont share the private keys rather it will store and use those keys for decryption[1, 18].

1.5.2 Payment

In this phase the merchant gets the e-coin. Customer will purchase some goods, create e-coin and gives to merchant[1, 18].

1.5.3 Deposit

In this phase the merchant deposits the e-coin in the Bank. So that the amount of purchase will be deducted from the customer's account and credited to merchant's account.

1.5.4 Tracing

In this phase the e-coin is traced whether it is ready to use or not by customer.

1.6 Applications

e-cash systems replace the traditional system. Payments less than "one" is possible through e-cash systems. e-cash systems are used in online banking systems; they are also used for online shopping and other purposes. There are many features those can be achieved by e-cash System such as[1]

- Transferability: Electronic coins can be circulated among people regardless of whether the transactions are online of off-line.
- Portability: The security and use of digital cash is not dependent on any physical location. The cash can be transferred through computer networks into storage devices and vice versa
- Off-line Payment: The transaction can be done off-line, meaning no communication with the central bank is needed during the transaction.
- Unforgeability: Only authorized parties (i.e. the bank) can produce digital coins.

- Anonymity: The spender of the cash must remain anonymous. If the coin is spend legitimately, neither the recipient nor the bank can identify the spender.
- Unreusability: The digital cash cannot be copied and reused. Then we have to minimize the risks for forgery and establish a good authenticity system.
- Divisibility: Digital cash can be divided into smaller amounts.

1.7 Preliminaries

1.7.1 Discrete Logarithm Problem

Let 'G' be a cyclic group of order 'q' with a generator 'g'. Equivalently, for every h belongs to G, there is a unique X belongs to Z_q such that g^x=h and x is called the discrete logarithm of h with respect to g. The discrete logarithm assumption states that there exists a group G such that computing the discrete logarithm is hard and hence we have the discrete logarithm problem (DLP for short) [6]. In simplest form let y=g^x mod(n), if g, n and x are given we can easily find out y, but if y, g and n are given then it is very hard to find out x. This is Discrete Logarithm problem.
for example: x=9^{2*i+1} mod 10 and y=9^{2*i} mod 10 , where i=0,1,2,...
If i is given then we can easily evaluate x or y but if x or y is given then it is very difficult to find i[19].

1.7.2 Integer Factorization Problem

Integer Factorization as a Decision Problem, given two integers A, k. Does there exist a prime number p such that 2 < p < k and p completely divides A ?[7]
"YES" instance => we can find a prime number p that satisfies the above requirements
"NO" instance => we cannot find any prime number that satisfies above requirements.
Clearly Integer Factorization is in NP. Integer Factorization lies in NP, but we don't know exactly how hard it is. In simplest form let n=p*q, if p and q are given then we can find n easily but if n is given to find out p and q is very hard. This is Integer Factorization Problem.
for example: 100 can be factorized as 20*5, 10*10, 5*5*4, 5*5*2*2, etc. so we can't say they should factorize this way.

1.7.3 Safe Primes

These primes are called "safe" because of their relationship to strong primes. A prime number is a strong prime if $q+1$ and $q-1$ both have large prime factors[8]. For a safe prime,$q = 2 * p * f + 1$, the integer p is a large prime factor. Here q always can't be prime for example if p=f=2, q=9 is not a prime. So we have to test whether it is prime or not. If q is a safe prime then the order of multiplicative subgroup will be very large.

1.8 Objective

The objective of ours is to produce an e-cash scheme which satisfies all the properties like double spending detection, anonymity, unforgeability, unreusability, tracing and divisibility; this scheme will facilitate online payment and will mantain the above features. Moreover the proposed scheme attaches expiration date to coins so that the banking system can manage its databases more efficiently. The coins produced by the scheme can be transferred through computer networks into storage devices and vice versa so that portability is assured.

1.9 Contribution

We have proposed an offline scheme which is untraceable, maintains the double spending, unforgeability; and maintains portability, divisibility. Moreover, it can withstand CCA(chosen cipher text attacks).

1.10 Outline of Thesis

The protocol of e-cash system is proposed in chapter 2. All the implementation details are described in chapter 3. All the security analysis are described in chapter 4. we propose the conclusion in the chapter 5.

Chapter 2

Proposed offline e-cash Protocol

The e-cash system consists of 4 participants :

- Customer: Customer buys goods or stuffs and gives e-coin to merchant rather giving cash or cheque.
- Merchant: Merchant takes e-coin from the customer and deposits into the bank where the purchase amount will be deducted from customer's account and credited to merchant's account.
- Bank: Bank tracks all the transaction between merchant and customer, all the details of customer and merchant.
- Clearing House: Clearing House is a third party, it publishes the public keys and private keys for digital signature and cryptography.

The e-cash system also consists of 4 protocols :

- Account Opening Protocol: A new customer is added by taking a secret membership key i.e W.
- Payment Protocol: A customer buys some e-coin from bank and gives merchant so that he can purchase some goods.
- Deposit Protocol: A merchant deposits e-coin in the bank. Purchase amount will be deducted from Customer's account and credited into the merchant's account.
- Tracing Protocol: A customer can trace an e-coin whether it has been used or not.

Before discussing the protocols we should do some system setups.

2.1 System Setup

In this protocol, bank and clearing house create the public keys and private keys. They publish the public key publicly. These are the following steps: [1, 17]

Step 1: Bank randomly chooses 3 prime numbers p', q' and f such that $p = (2{*}p'{*}f{+}1)$ and $q = (2 * q' * f + 1)$ are primes.

Step 2: Chooses e which is co-prime with both $p-1$ and $q-1$.

Step 3: Computes $n = p * q$, ϕ(n)=$(p-1) * (q-1)$ and d=e^{-1} mod ϕ(n).

Step 4: Chooses g as a generator of n of order f.

Step 5: Clearing house chooses x in $Z_{\phi(n)*}$ and computes Y=g^x mod n.

Step 6: Publishes Public key(Y, g, n, e) , Private key (d, p, q, f, x).

Step 7: In Digital Signature signer signs with private key and verifier verifies with the public key.

2.2 Account Opening Protocol

It involves protocol between the customer and the bank in which the customer gets the secret membership key.

Step 1: The customer chooses random number a,α,β in Z_{n*} and computes
δ=α*β mod(n)
Z=a*δ mod(n) and sends (δ,Z) to the bank.

Step 2: The bank computes w=Z * δ^{-1} mod n, sends w to the customer and stores (w, Z) with the customer's identity.

Step 3: The customer checks the authenticity by checking $a \equiv w$ mod n and keeps (w, δ,Z) as his secret membership key.

2.3 Payment Protocol

It involves protocol between the customer and the merchant in which the customer pays e-coin to merchant.

Step 1: The merchant generates a payment message M as follws and sends it to the customer. $M = H(Shopid, amountofpurchase, date)$.

Step 2: The customer requests a timestamp to the clearing house and obtains a timestamp t and chooses random number u in $Z_{\phi(n)*}$ and sends u, t.

Step 3: The customer computes
$S{=}Y^e \bmod(n)$
$R_1{=}S + M{*}g^u$ mod(n) and sends (S, R_1 , t) to the clearing house.

Step 4: The clearing house checks the authenticity by verifying $S^d = Y mod(n)$. If it satisfies, then chooses a random number K in $Z_{\phi(n)*}$ and computes
$R_2 = (k- \ R_1 \ -t)$ mod ϕ(n)
$R = (\ R_1 \ -S) \ g^{-k}$ mod n
$S + l \ = \ x \ - \ 1(\ R_2 \ -u+ \ R_1 \ ) mod \ \phi$(n) and sends $(S + l, R, \ R_1 \ , \ R_2 \ , t)$ to the customer.

Step 5: The customer computes l from the $S+l$ and selects μ in random and computes $\zeta \ = \ g^{\mu x}$ mod (n) and $\psi{=}w{*} \ Y^{\mu}$ mod (n) and sends the e-coin($t, S, R, l, \ \zeta$,ψ) to merchant.

Step 6: The merchant verifies the e-coin as per following condition $M = R{*} \ Y^{(S+l)} \ g^t$ mod n. If so merchant accepts the e-coin otherwise it is rejected

2.4 Deposit Protocol

It involves protocol between the merchant and bank in which the merchant deposits the e-coin to the bank.

Step 1: The merchant sends the e-coin to the clearing house and the clearing house computes $w{=}\psi/\zeta$ and sends w to the bank.

Step 2: Based on the identity w ,the bank determines the real identity of a customer. Then deducts the money from the customer's account and credits to the merchant.

2.5 Tracing Protocol

It involves protocol between the customer and bank where customer needs to know whether the e-coin has been used or not.

Step 1: Each e-coin is represented by 6-tuples and all tuples are uniquely define one e-coin. So $(t, S, R, l, \zeta, \psi)$ are the parameters of e-coin.

Step 2: Let E_1 be an e-coin(t_1,S_1,R_1,l_1,ζ_1,ψ_1) and E_2 be an e-coin(t_2,S_2,R_2,l_2,ζ_2,ψ_2).

Step 3: For E_1 and E_2 to be same all 6 tuples are to be same. t_1=t_2 can be possible if they are generated at same time. S_1=S_2 can be possible as they depend on Y,*i.e.* R_1=R_2 can be possible if different customers buy same amount of purchase from same merchant. (ζ_1,ψ_1)= (ζ_2,ψ_2) can be possible if they are same customers. As same merchant can't buy from same merchant at same time.So E_1 is always unique.

Step 4: After e-coin is used the tuples are stored in a database of used e-coin with date, merchant name and customer name.

Step 5: When a customer comes to trace e-coin then the e-coin is matched against all the used e-coins by the customer.

Step 6: If match is found then it will show that this e-coin has been already used at this date, by this merchant.

Step 7: If match is not found then it will show that this e-coin has not been used.

2.6 Correctness of Protocol

There are two proofs of correctness

- Given: $S = Y^e mod(n)$
  To prove: $S^d = Y mod(n)$
  We know $d = e^{-1} mod\phi(n) => ed = 1 mod\phi(n) => ed = k * \phi(n) + 1$
  $S = Y^e mod(n)$
  $=> S^d = Y^{ed} mod(n)$
  $=> S^d = Y^{k*\phi(n)+1} mod(n)$
  $=> S^d = Y mod(n)$

- To prove: $M = R * Y^{(S+l)} * g^t mod n$.
  Proof: $R * Y^{(S+l)} * g^t mod n$
  $= R * g^{x(S+l)} g^t mod n$
  $= R * g^{(R2-u+R1) mod \phi(n)} * g^t mod n$
  $= (R1 - S) * g^{-k} * g^{(R2-u+R1) mod \phi(n)} * g^t mod n$
  $= M * g^u * g^{-k} * g^{(R2-u+R1) mod \phi(n)} * g^t mod n$
  $= M * g^{(R2-k+R1+t) mod \phi(n)} mod n$
  $= M * g^{a*\phi(n)} mod n = M mod n$, where a=0,1,2...

2.7 Why this protocol is called offline

Let us discuss the Payment protocol. Initially Bank and clearing house generates the public keys and private key which are required for all protocols.

Step 1: The customer purchases from the merchant and the merchant generates M.

Step 2: Now the customer requests clearing house for timestamp and gets it.

Step 3: Now the customer signs on the message. He sends both the sign and the message.

Step 4: Now the clearing house verifies and sends some components of the e-coin.

Step 5: The customer creates an e-coin, which contains 6 tuples and sends the e-coin to merchant.

Step 6: The merchant verifies the e-coin using verifying algorithm.

So in this whole process, the bank remains silent. There is no role of bank during payment, that is why this e-cash system is called offline e-cash system.

Chapter 3

Implementation Detail

3.1 Introduction to Implementation

Implementation contains three base classes. They are Project class, Cash class and Deposit class. Project class initializes the public keys and private keys, opens account of customer, modifies customers and deletes customer. Cash class gives opportunities to the customer to buy the new e-coin, trace e-coin and view details of the customers. Deposit class gives opportunities to the merchants to deposit the e-coin. The detail implementations are given below :

3.2 Project Class

When we run this class a stylish frame will be generated as Figure 3.1 . After 3 seconds, a Login page will be generated as Figure 3.2 . Bank Administrator will use this software so that he can enter the corresponding username and the password. If the username and password are valid and they match with the username and password pair in the database, then the admin is accepted and the home page is generated. The home page will be like Figure 3.3 . This will provide different operations to admin such as: generating the public keys and private keys, adding new customer, deleting customer, modification of the customer details and tracking of all the customer details. This class uses key.properties, wname.properties and wbalance.properties files to store public keys, private keys, the customer details(each customer contains w(secret key), name of the customer, balance of the customer).
Now let us elaborate how different operations are done.

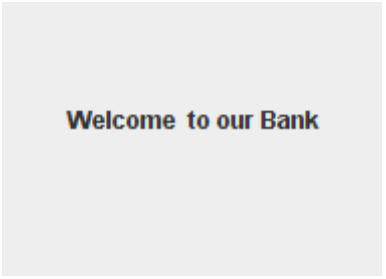

FIGURE 3.1: Welcome bank page.

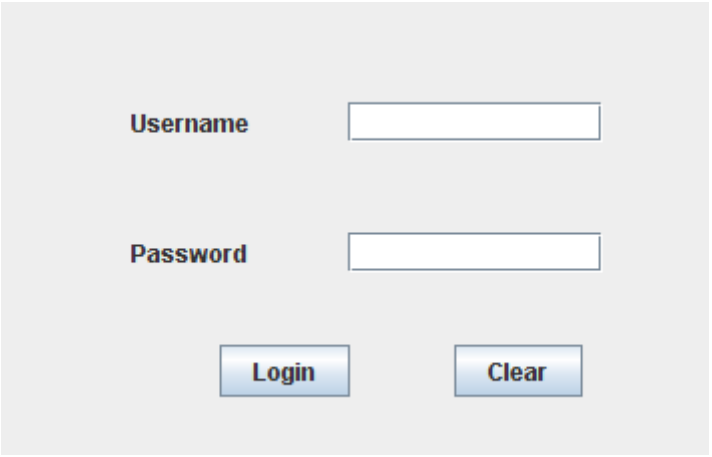

FIGURE 3.2: Login bank page.

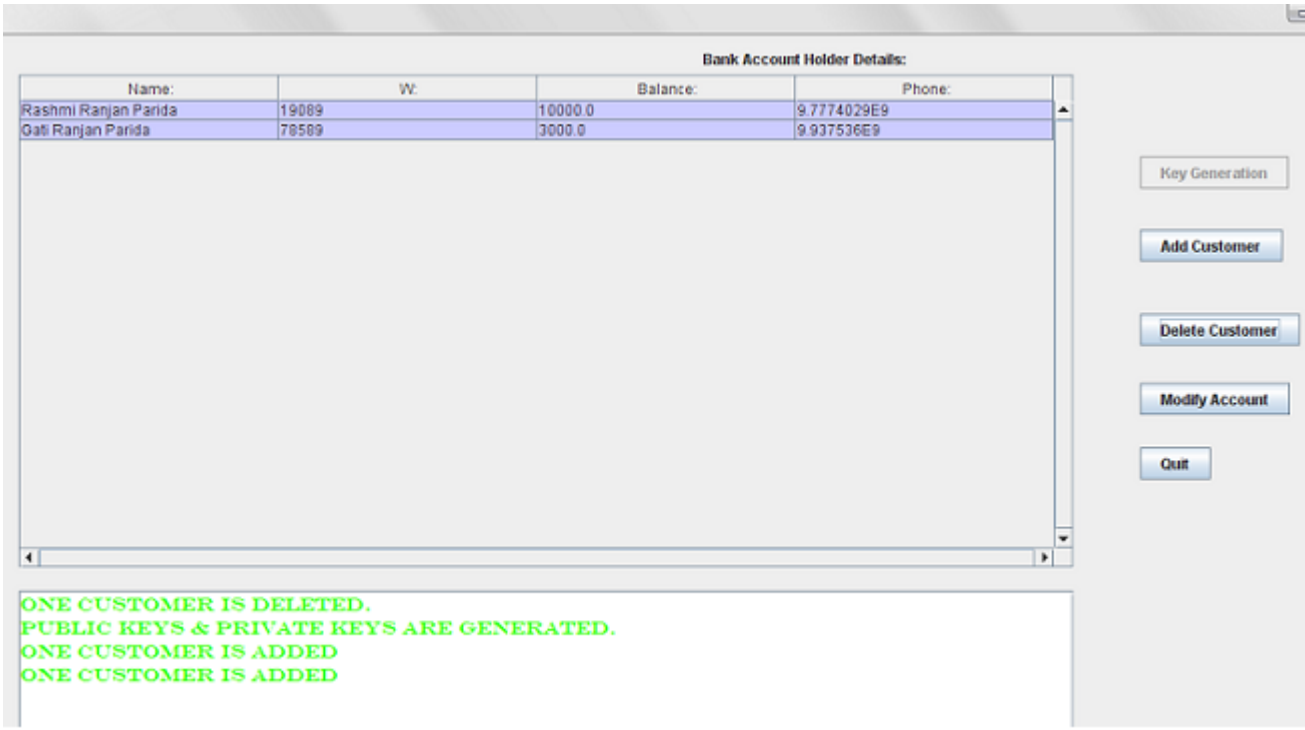

FIGURE 3.3: Home page for the admin.

3.2.1 Generation of Keys

The admin will click on a Key Generation button which makes the private and public keys to be generated. They are stored in key.properties file. All the keys' variables are initialized.

3.2.2 Add Customer

If the admin clicks on Add Customer button, a new form will be generated where all the details of customer will be filled and described as Figure 3.4. W *i.e.* a secret member

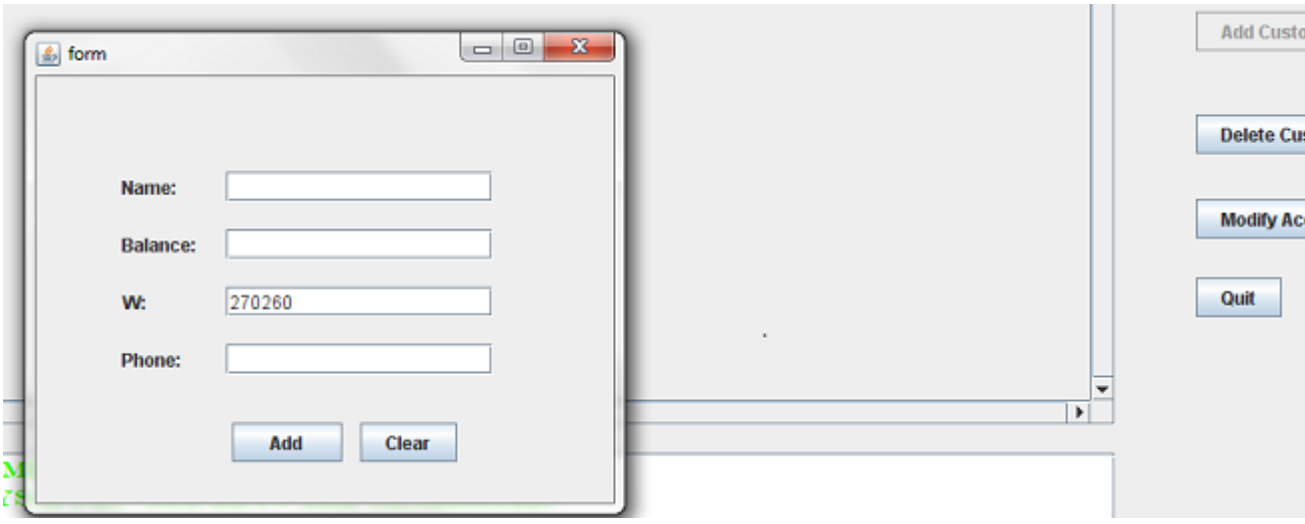

FIGURE 3.4: Add Customer for the admin.

key for a customer is generated and that is automatically filled. After clicking on add button a new record of customer is added into database and to the table. If balance or mobile field is filled with not integer, then an error frame is generated as described in Figure 3.5.

3.2.3 Delete Customer

If the admin wants to delete some customers, then he should click on the customer row which he wants to delete. After he clicks on the Delete customer button, the corresponding customer will be deleted from the table and database. If he clicks on Delete Customer before selecting any row, then an error frame will be generated as described in Figure 3.6.

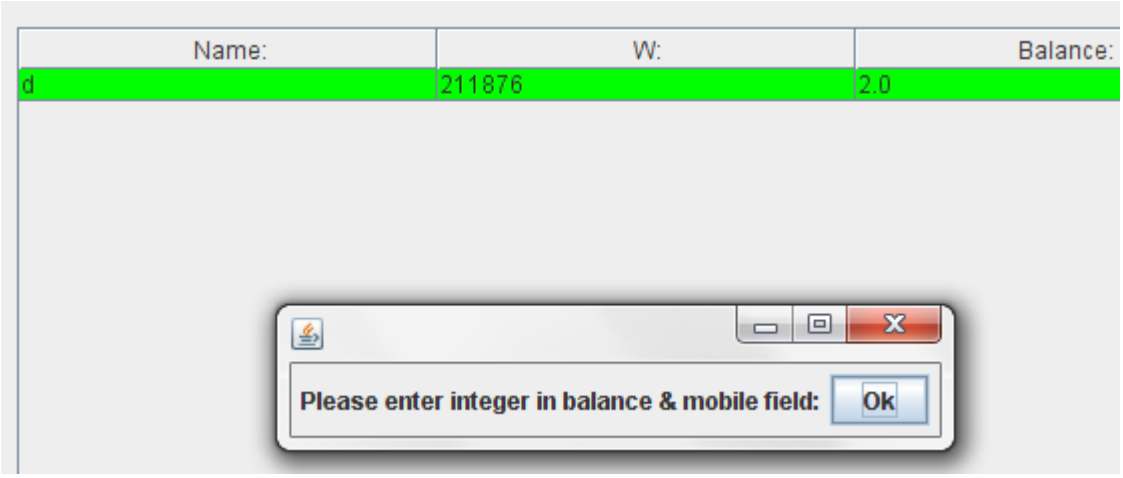

FIGURE 3.5: Enter valid balance or mobile.

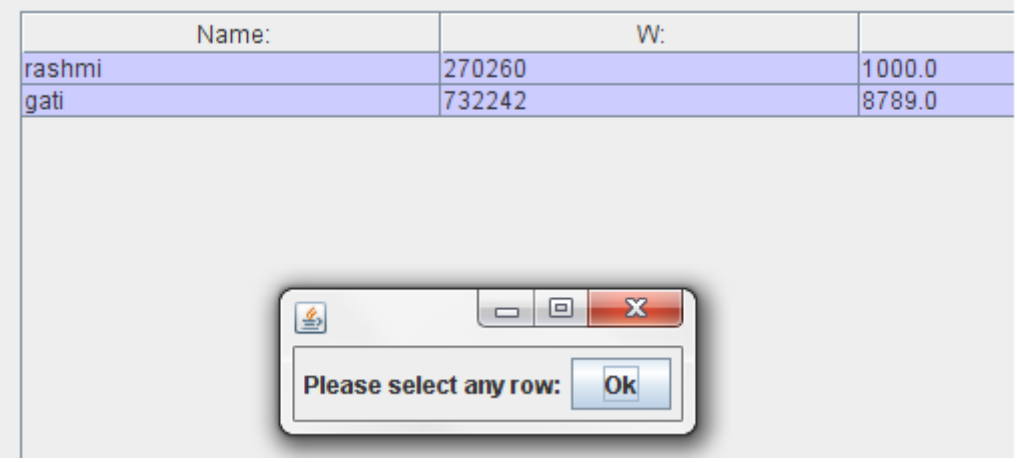

FIGURE 3.6: Select any row for the admin.

3.2.4 Modify Customer

If the admin wants to modify some details of customer, then he has to click on the customer row which he wants to modify. After clicking on Modify customer button, a new form will be generated as in Figure 3.7 where all the details will be given and the admin can change some details and after clicking on modify button, all the details of customer are changed in the database and the table. If he clicks on Modify Customer before selecting any row, then an error frame will be generated as described in Figure 3.6.

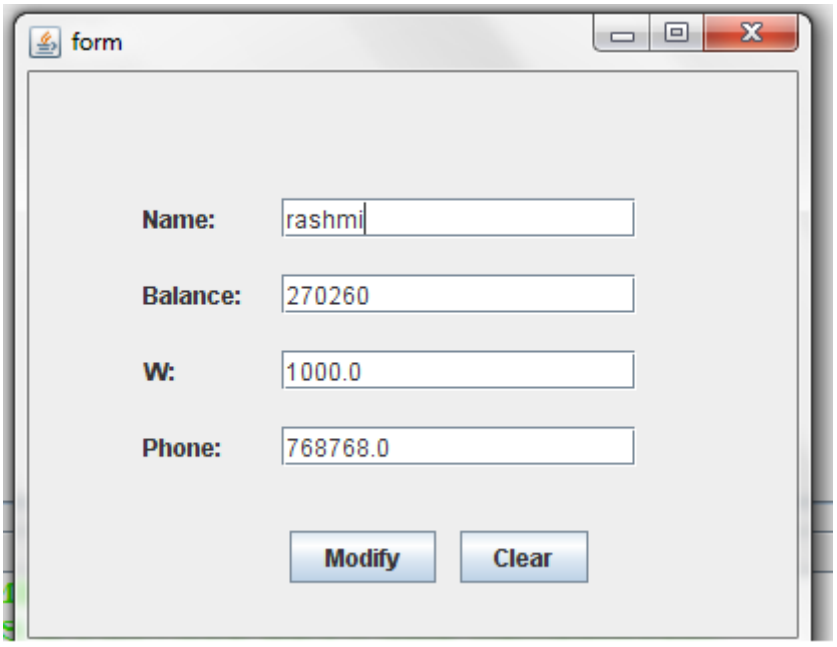

FIGURE 3.7: Modify Customer for the admin.

3.3 Cash Class

When we run this class a stylish frame will be generated as Figure 3.8 . After 3 seconds, a Login page will be generated as Figure 3.9 . Customer will use this software so that he can enter the corresponding username and password. If the username and password are valid and they match with the username and password pair in the database, then the customer is accepted and the home page is generated. The home page will be like Figure 3.10. This class will provide operations for customer such as buy e-coin, trace e-coin and view details.

3.3.1 Buy e-coin

In order to buy e-coin customer has to click on the merchant from whom he wants to buy and enter the amount of purchase in buy field and click on buy button. A property file will be generated which contains all the values of e-coin as described as Figure 3.11. If he clicks on Buy button before selecting any row, then an error message is generated showing that you have to select any row. If the balance is less than the purchase amount, then also an error message is generated describing less balance.

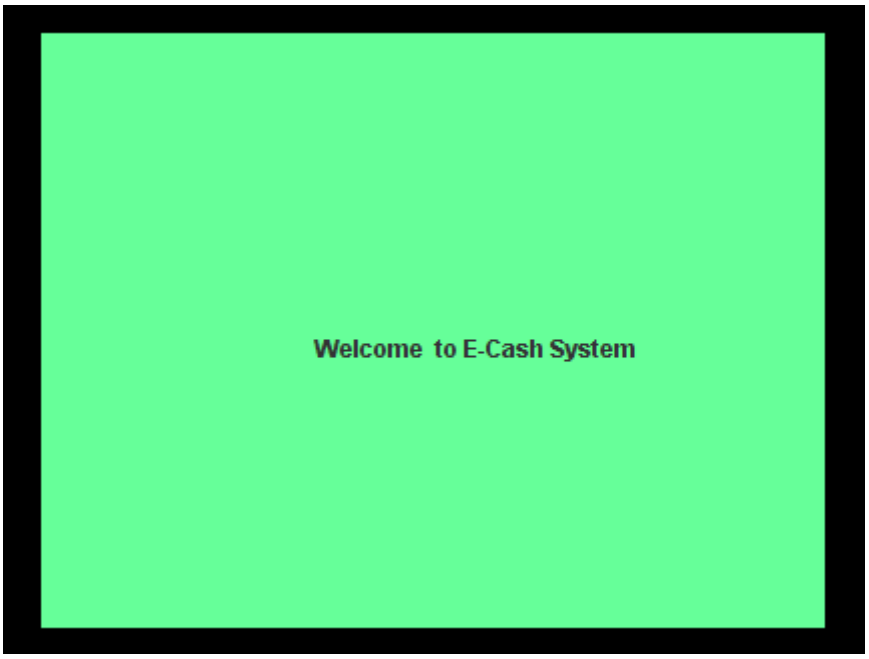

FIGURE 3.8: Welcome Page for the Customer.

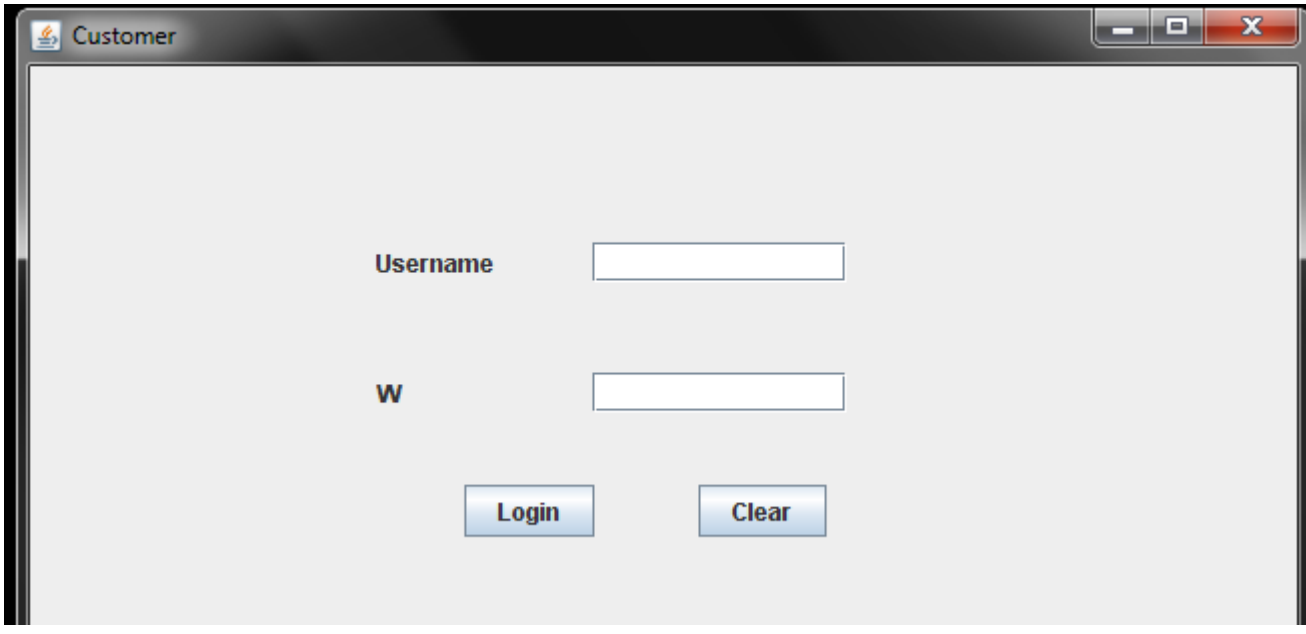

FIGURE 3.9: Login page for Customer .

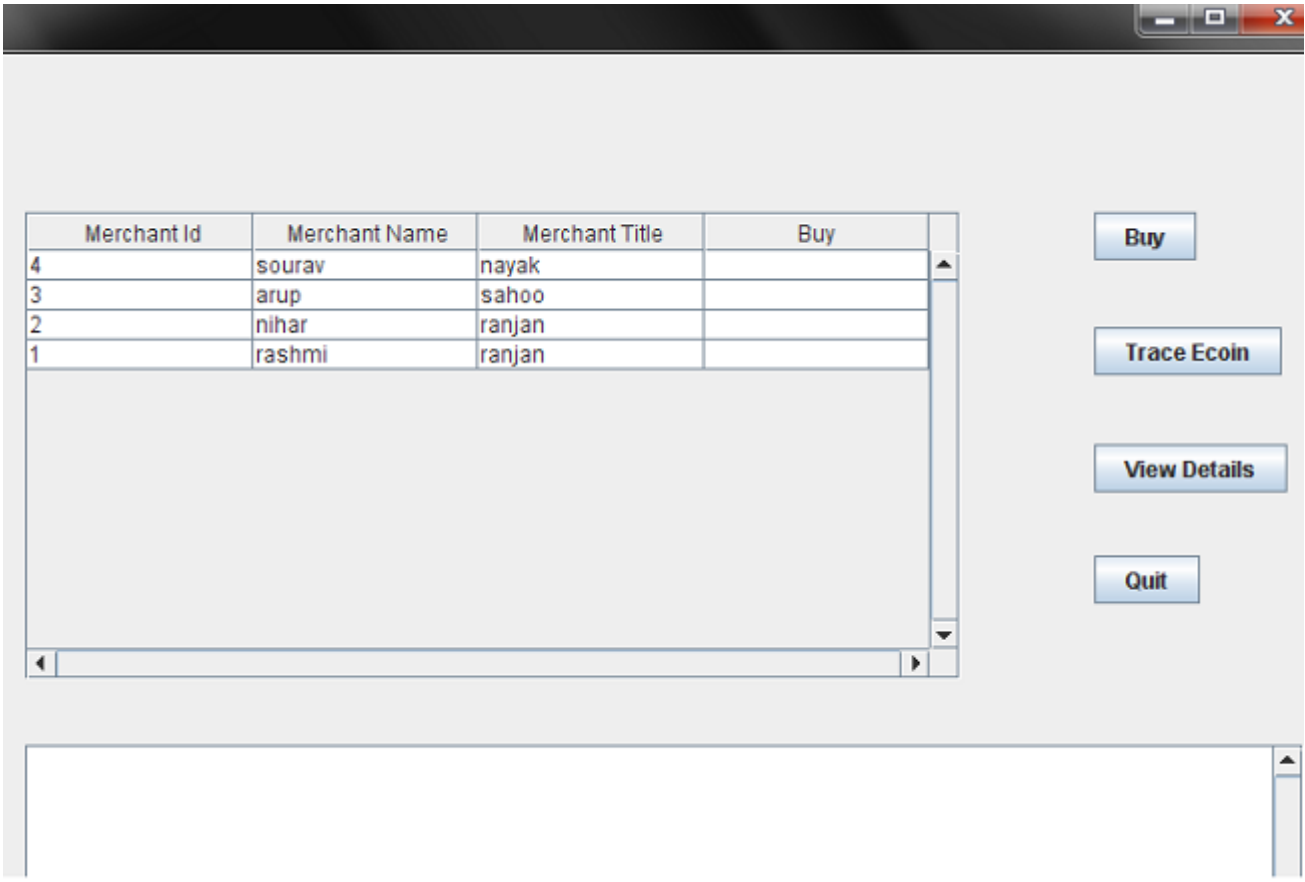

FIGURE 3.10: Home Page for the Customer.

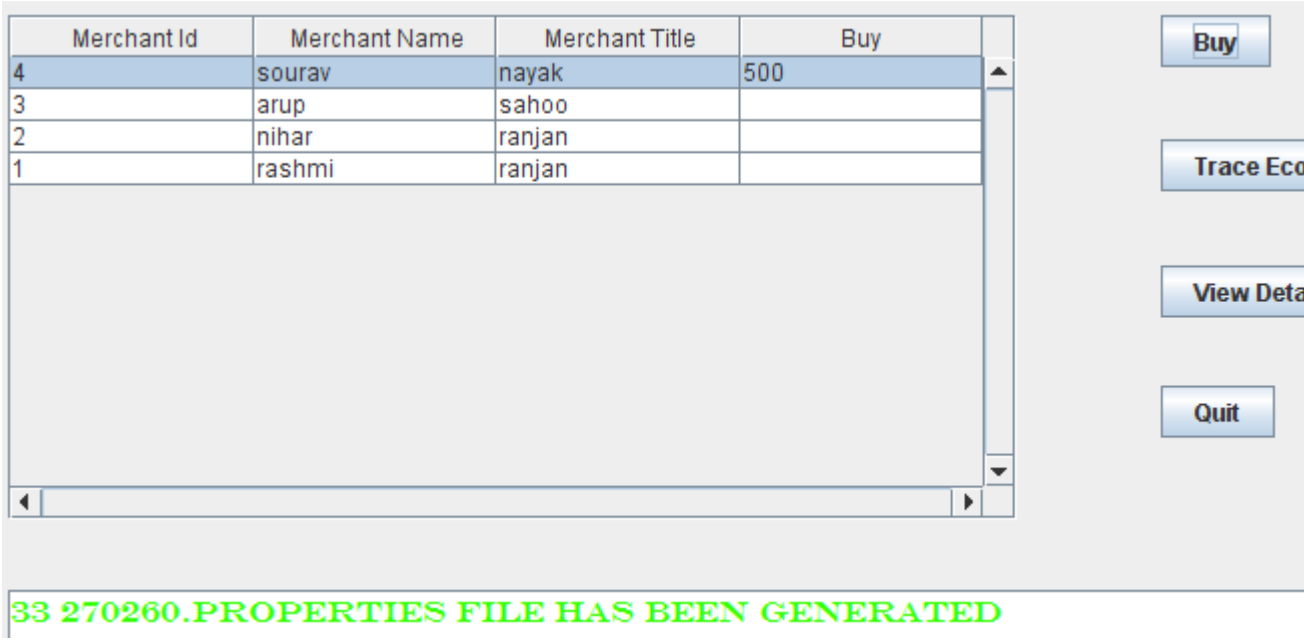

FIGURE 3.11: Buy e-coin Page for the Customer.

3.3.2 Trace e-coin

In order to check whether an e-coin is used or not Customer can trace an e-coin by clicking on Trace e-coin button. Then a new file chooser will be generated as Figure 3.12. Then it will show whether the e-coin can be used or already used.If the e-coin has already been used, then an error frame will be displayed as described in Figure 3.14. If the e-coin has not been used, then a frame will be displayed as described in 3.13.

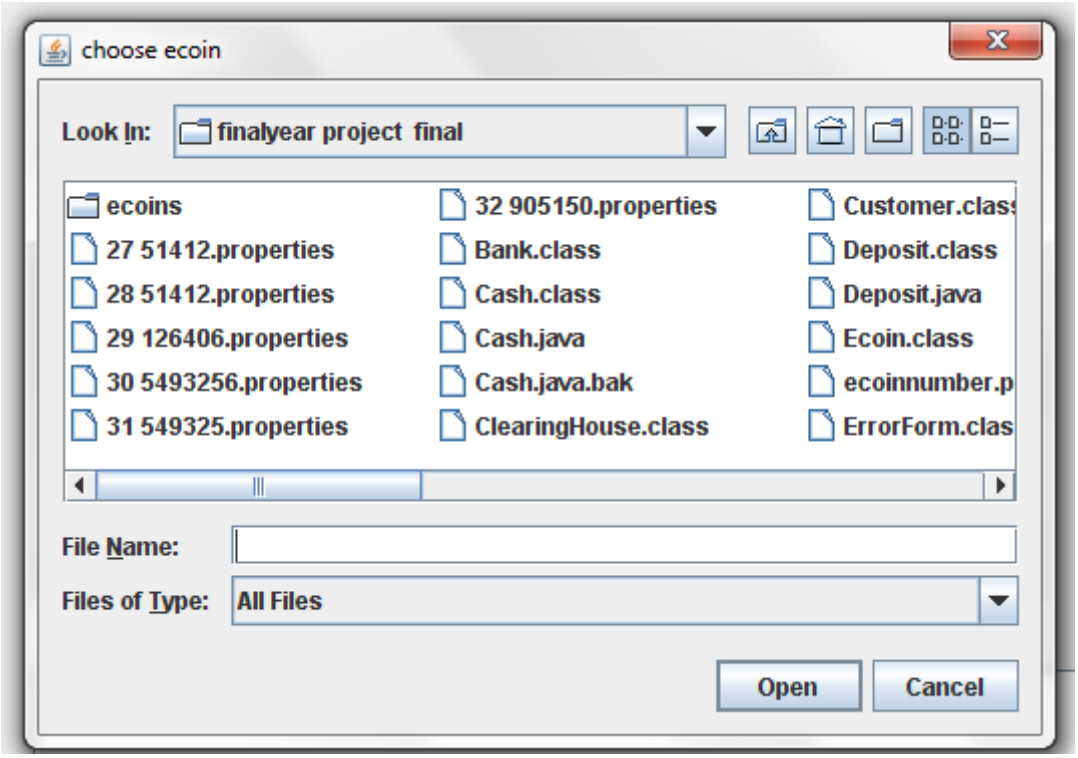

FIGURE 3.12: Trace e-coin Page for the Customer.

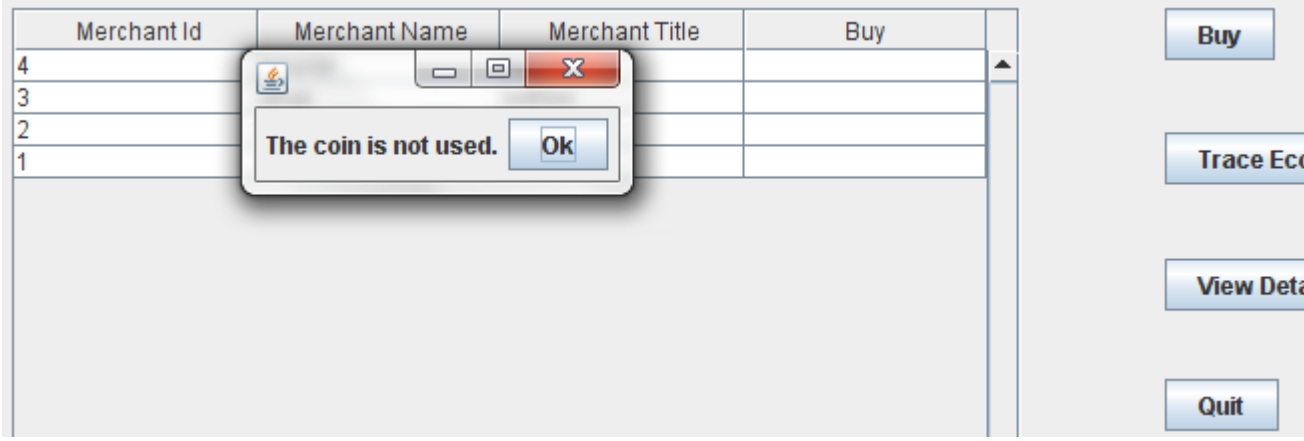

FIGURE 3.13: Trace e-coin Page for the Customer.

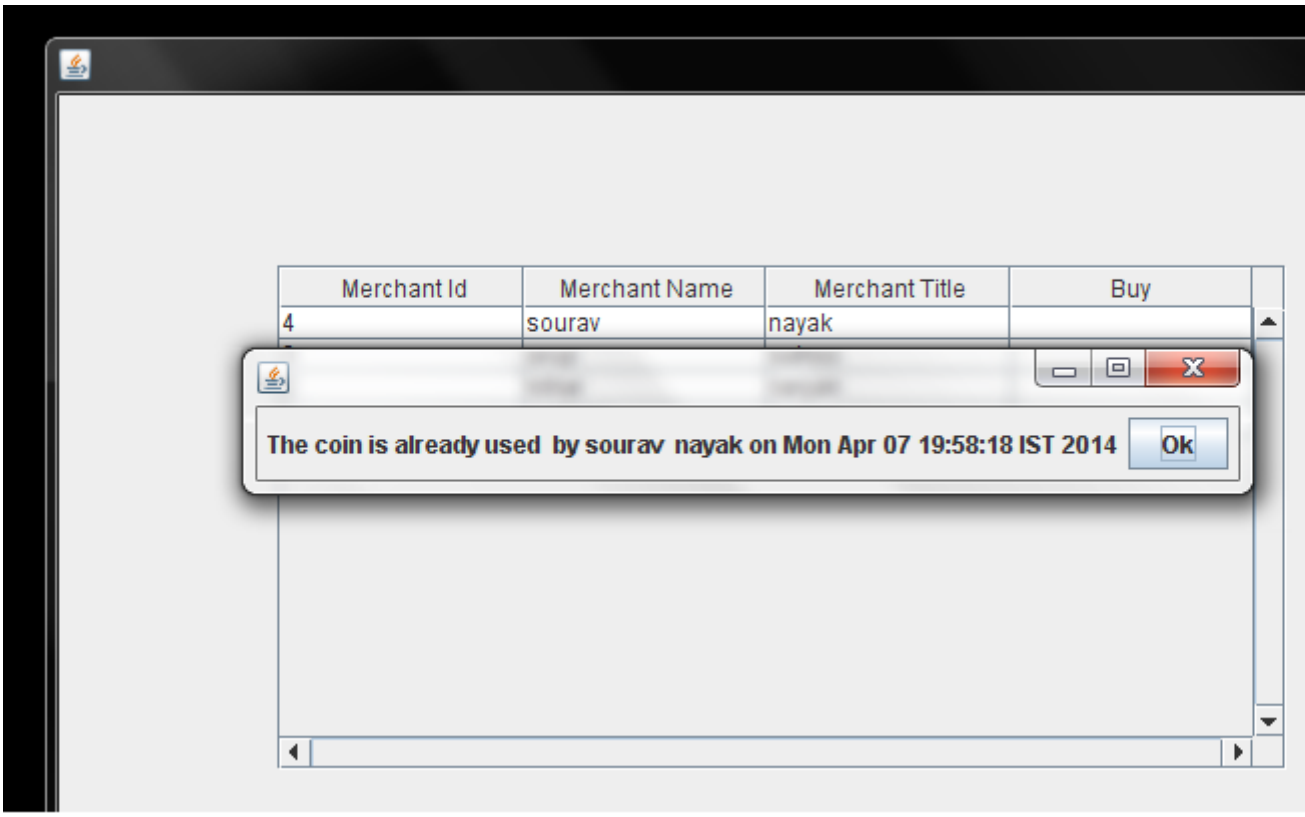

FIGURE 3.14: Trace e-coin Error for the Customer.

3.3.3 View Details

In order to check the balance, customer will click on view details button after which a separate window will be generated as Figure 3.15. After checking balance, he can click on ok button to make the frame to disappear.

FIGURE 3.15: View Details Page for the Customer.

3.4 Deposit Class

When we run this class a stylish frame will be generated as Figure 3.16 . After 3 seconds, a Login page will be generated as Figure 3.17 . Merchant will use this software to enter corresponding username and password. If the username and password are valid and they match with the username and password pair in the database, then only the merchant is accepted and the home page is generated. The home page will be like Figure 3.18. This class will provide operations for merchant such as deposit e-coin. He will deposit e-coin by clicking on Deposit button, then a new form will be generated as described as Figure 3.19. If he clicks on Deposit button a file chooser will be generated as described as Figure 3.20. He chooses the properties file which he wants to deposit. If the e-coin has not been used, then it will decoded and amount of purchase will be added to the merchant balance and a copy of e-coin is added to e-coin database. If the merchant is trying to use same e-coin multiple times then a "double spending error" is generated using e-coin database as described in Figure 3.21. If the merchant steals an e-coin from other and tries to deposit, then also a "you are not correct merchant error" is generated using e-coin database as described in Figure 3.22. If the merchant changes the name of the e-coin file, then also double spending is detected as described in Figure 3.21.

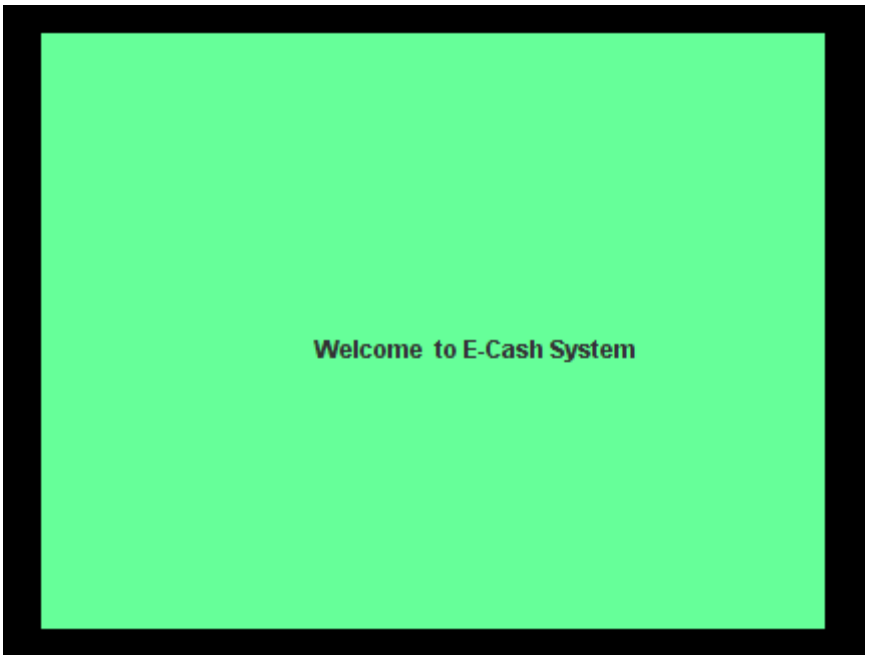

FIGURE 3.16: Welcome Page for the Merchant.

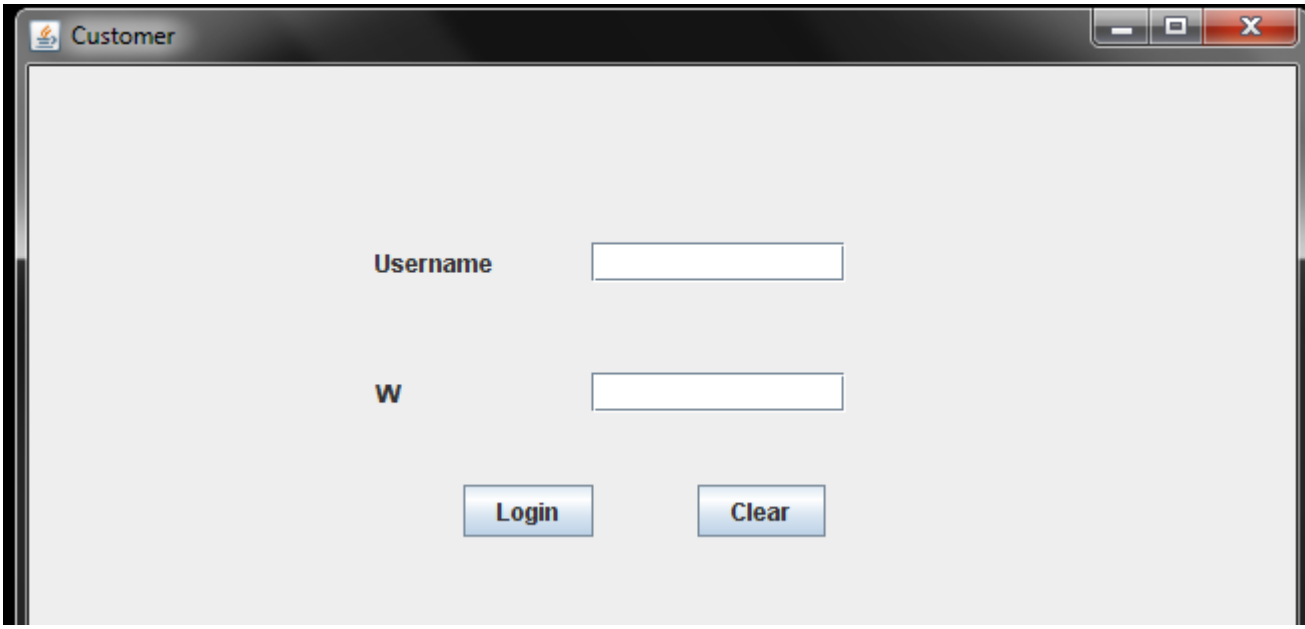

FIGURE 3.17: Welcome Page for the Merchant.

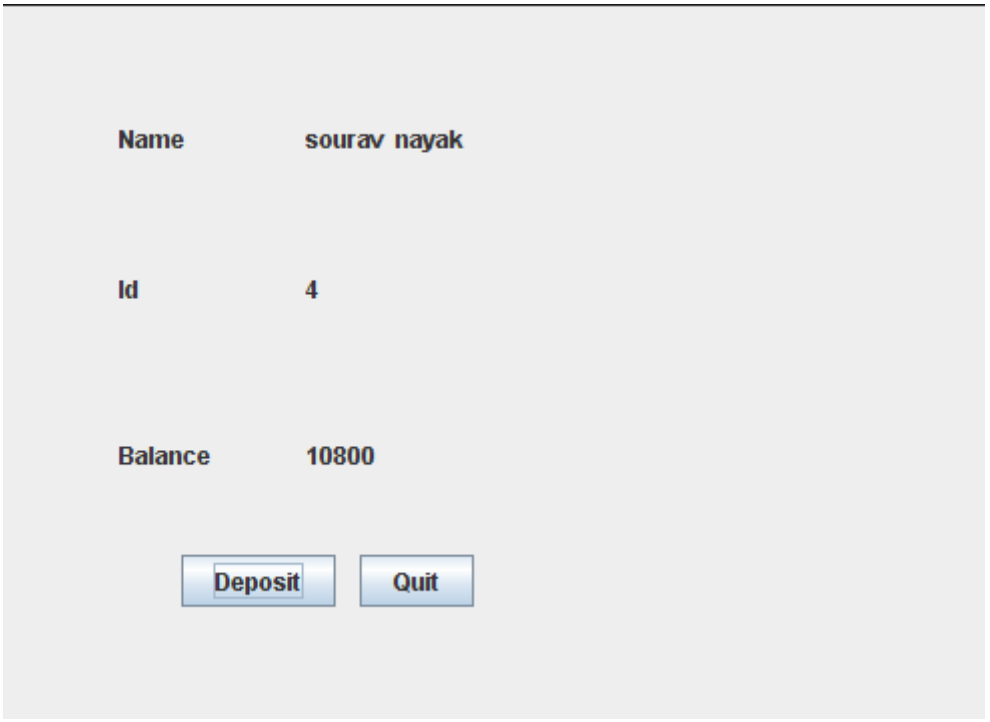

FIGURE 3.18: Merchant Page for the Merchant.

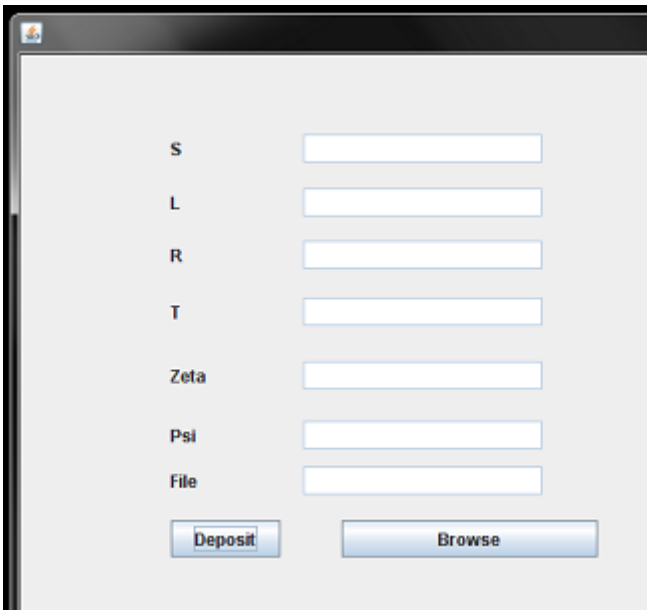

FIGURE 3.19: Deposit e-coin for the Merchant.

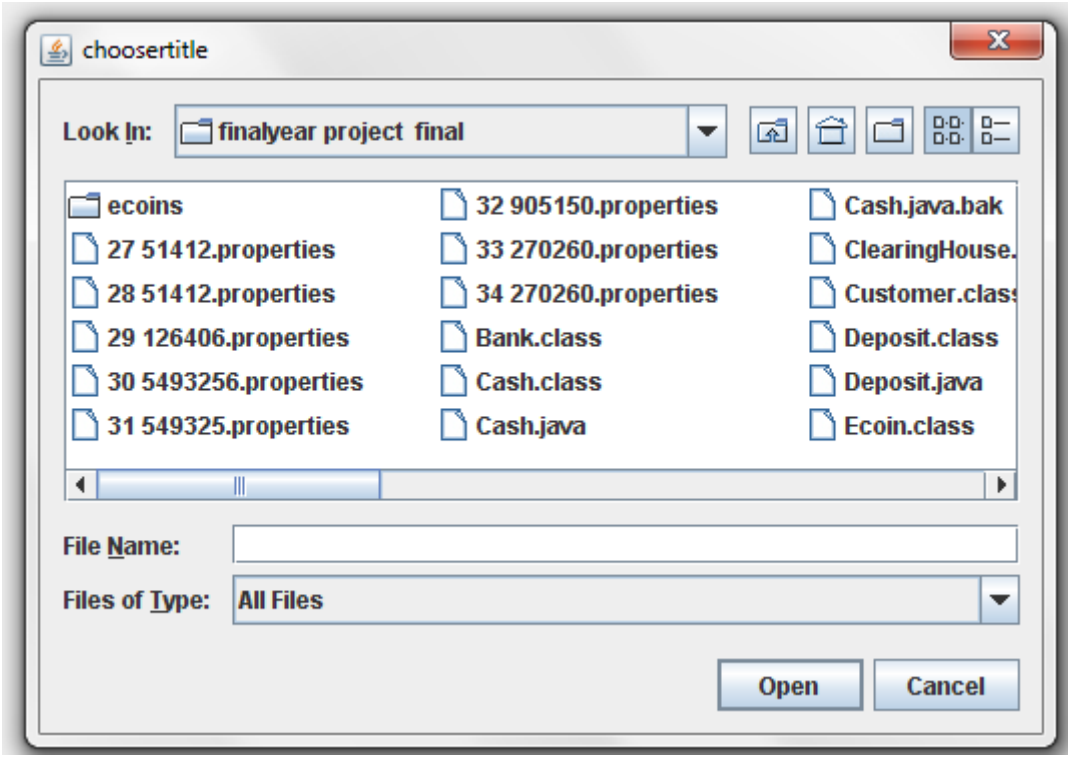

FIGURE 3.20: Choose e-coin for the Merchant.

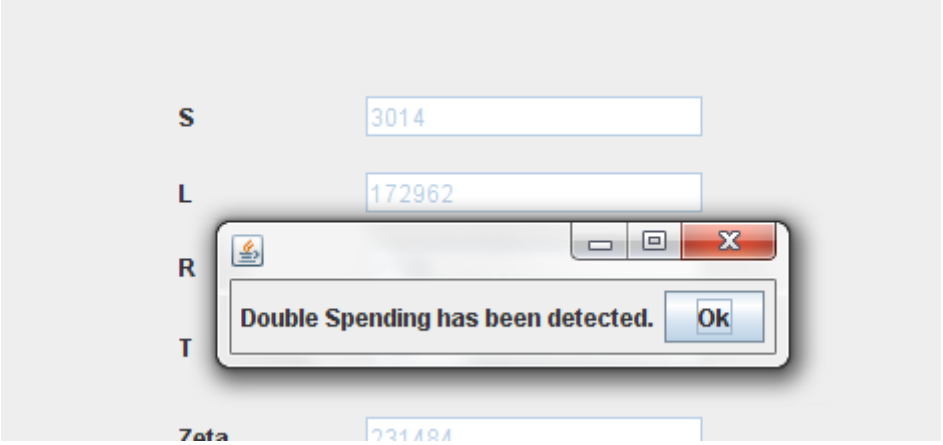

FIGURE 3.21: Double Spending for the Merchant.

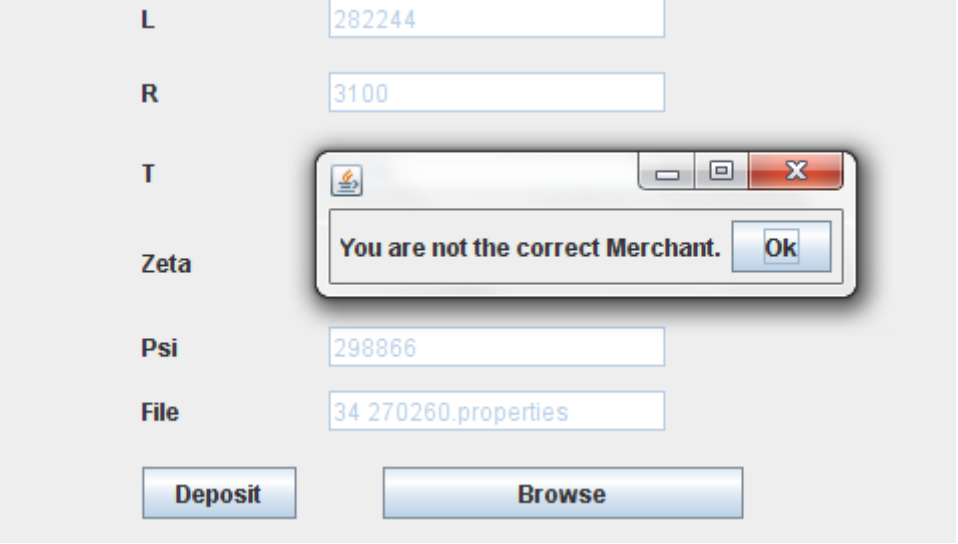

FIGURE 3.22: Not Correct Merchant.

Chapter 4

Analysis of Proposed e-cash scheme

In an offline e-cash system, analysis of security features is one of the foremost jobs, which needs to be carried out so to have better payment facilities between a payer and payee; the analysis of e-cash system is done so that the e-cash can withstand the security features like: unforgeability, double spending, anonymity and can achieve untraceability. e-cash system is the way or the medium in which various protocols are implemented to facilitate the payments between the consumer and the merchants. Merchants also can deposit the e-coin to the bank and receive the payments from the customers. The analysis is done so that it can ascertain the security flaws in a fair e-cash system. Moreover the improved scheme after analysis also is resistant against threats and attacks. Anonymity, unforgeability and double spending detection are the most important security features[1]. Our design of e-cash system should be such that it can withstand all the attacks from the malicious programmers. However, since authorities such as banks and certificate authorities may have important secret data of customers, the insiders in the potentially untrusted authorities can become threats to electronic cash systems.

4.1 Security Features

The security features are broadly described below:

4.1.1 Anonymity

We execute the withdrawal protocol for spender S. The messages between the Spender S and the Bank B is called Trans. A random bit b is chosen. If b=0, then a random valid

coin coin0 is created. A is given coin, b , transcript and all the identity information of S that the Bank has. A then output a bit b. The output is either a bit b or 0 otherwise. A adversary has the access to all the exchanged message between the Spender and the Bank. We say that P achieves anonymity if the adversary succeeds with probability that is at most negligibly greater than 1/2. To prove this, we first note that $Pr[b = 0] = Pr[b = 1] = 1/2$. Coin has three attributes: Trans, Ids, coinx. Using the three attributes, the adversary can identify if the coin is actual or random. Distinguishing between these cases is equivalent to linking a coin to the spender and violating anonymity[1].
$Pr[ExpA] = 1/2 * Pr[ExpA]b = 1 + 1/2 * Pr[ExpA]b = 0$
$= 1/2 * Pr[A(Trans, Ids, coin1) = 1] + 1/2 * Pr[A(Trans, Ids, coin0) = 0]$
$= 1/2 * Pr[A(Trans, Ids, coin1) = 1] + 1/2 * (1 - Pr[A(Trans, Ids, coin0) = 1]$
$<= 1/2 + 1/2.Pr[A(Trans, Ids, coin1) = 1]–Pr[A(Trans, Ids, coin0 = 1]$
Hence, all values of b1 and l occur with the same probability and this means
$Pr(A(Trans, Ids, coin1 = 1)] - Pr[A(Trans, Ids, coin0) = 1 << \epsilon$ where ϵ is negligible.
$Pr[ExpA] <= 1/2 + \epsilon$

Note that the numbers $\beta 1$ and $\beta 2$ are very important, since using the coin once does not allow identification of the Spender, but using it twice does. To see the effect of $\beta 1$, $\beta 2$ suppose b1 is essentially removed from the process by taking $\beta 1$=1. Then the Bank could keep a list of values of c, along with the person corresponding to each c. When a coin is deposited, the value of H would then be computed and compared with the list. Probably there would be only one person for a given c, so the Bank could identify the Spender[1, 9, 10, 12].

4.1.2 Unforgeability

Unforgeability means that the e-coin can only be generated by the Bank itself. We can't make or forge an e-coin by our own. If we want to make an unauthorized coin we need to have α^r=Az^H(u,g,A). But this is a case of Discrete Logarithmic Problem which cant be solved. So our e-cash scheme is unforgeable.[1, 11]

4.1.3 Double Spending Detection

Double spending detection is the mechanism of detecting if a coin is spent by the customer twice. Suppose a customer uses the coin for Merchant M and again uses the same coin for merchant N. Then it is the case of double spending detection; it is detected by the help of properties file of coins. A coin is represented by six tuples$(T, S, R, L, \delta, \psi)$;

all the information of the coin is stored in a properties file. Using the contents of this properties file, we can detect the double spending[1, 10].

4.2 Attacks

4.2.1 Chosen Ciphertext Attack

In chosen ciphertext attack, attacker knows the deciphering algorithm and uses that with many chosen ciphertexts. Comparing the results obtained from these cases, the attacker obtains the plaintext. There are two types of chosen ciphertext attacks:

- Adaptive chosen ciphertext attack: Adaptive chosen ciphertext attacks are the attacks in which attacker chooses a number of ciphertexts and obtains the corresponding plaintexts. Using the relationship between the plaintext and ciphertext pairs, we obtain the actual plaintext. So, adaptive chosen ciphertext attacks are quite dangerous attacks. In this attack, in order to find the exact contents of ciphertext 'c', the attacker, using the deciphering algorithm, finds out the plain text and ciphertext pairs. After that the attacker extracts the contents of the actual ciphertext object c by the help of information obtained from plaintext-ciphertext pairs.
- Indifferent chosen ciphertext attack: It is an attack model for cryptanalysis in which the cryptanalyst gathers information, at least in part, by choosing a ciphertext and obtaining its decryption under an unknown key. In the attack, an adversary has a chance to enter one or more known ciphertexts into the system and obtain the resulting plaintexts. From these pieces of information the adversary can attempt to recover the hidden secret key used for decryption.

There are three methods for immunizing, the first one is based on use of one-way hash functions, the second on the use of universal hash functions and the third on the use of digital signature schemes.

- Immunizing with one-way hash functions [13] : Here we immunize our e-cash system against the CCA attacks by using one-way hash functions; some predefined enciphering and deciphering algorithm are used for immunizing.
- Immunizing with universal hash functions [14, 15]: In this case, universal hash functions are used for immunizing against CCA attacks.
- Immunizing with Digital Signature Schemes [16]: Based on DSA and one way hash functions, we immunize against the CCA attacks.

4.2.2 Key-only attack

It is an attack in which the secret key e is only known. If N is known, then $X = C^e mod N$ So, X can be found easily. So plaintext X can be found easily if the secret key is known[2, 3].

4.2.3 Fogeary attack

This is used to forge signatures; strongly unforgeable signatures can be constructed based on Computational Diffie Hellman(CDH). Using CDH it is possible to achieve unforgeability [2, 3].

4.2.4 Man in the Middle Attack

It is an attack in which the another merchant N receives the payment from a customer P which is intended for merchant M[2, 3].

4.2.5 Impersonation Attack

Merchants M and N try to deceive the customer. The customer has spent 'x' amount but he is debited with 'y' amount by the merchants. One merchant impersonates other merchant [2, 3].

Chapter 5

Conclusion and Future Scope

In our new e-cash scheme, we propose a new off-line electronic cash system which maintains the security features like anonymity, double spending, unforgeability. It is also immune to various CCA attacks and can withstand them; it implements the tracing protocol. We use three different protocols: account opening, payment and deposit protocol for the development of the e-cash system which can be used for secure payment across internet and other networks. A secure e-cash system is the need of the hour which incorporates all the security features in it and helps in the secure payments to the merchants. It also takes care of the various attacks to which the e-cash is vulnerable and tries to resist them; In a nutshell, this e-cash scheme facilitates payments to the merchants and helps in exchange of merchandise also by implementing various protocols for secure communication between the merchants and the customers as well as between the merchants and the Bank.

In future research can be done on our scheme to lower its computation cost and communication overhead. Also research can be done to incorporate time-stamping feature to some of the highly proved secured signcryption schemes which can be applicable to highly security sensitive application like e-bidding, e-voting, etransactions etc.

Bibliography

[1] Ziba Eslami, Mehdi Talebi *A new untraceable off-line electronic cash system* Department of Computer Science, Shahid Beheshti University, G.C., Tehran, Iran,2011

[2] Behrouz A. Forouzan. *Cryptography and Network Security* . Tata McGraw-Hill, 2007.

[3] William Stallings. *Cryptography and Network security: Principles and Practices* . Prentice Hall Inc., 1999.

[4] W. Diffie and M. E. Hellman. *New directions in cryptography* . IEEE Transactions on Information Theory, 22(5):644-654, 1976.

[5] Masashi Une. *The security evaluation of time stamping schemes:* The present situation and studies, December 21 2001.

[6] T. ElGamal.*A public-key cryptosystem and a signature scheme based on discrete logarithms* . IEEE Transactions on Information Theory, IT-31:469-472, 1985.

[7] Paul C. van Oorschot Alfred J. Menezes and Scott A. Vanstone.*Handbook of Applied Cryptography* . CRC Press, 1996.

[8] S. Baral S. Mohanty, B. Majhi. *A novel time-stamped signature scheme based upon dlp* . In 1st International conference on Recent Advances in Information Technology (RAIT), pages 6-10, 2012.

[9] Chaum, D. *Blind signatures for untraceable payments* . In Crypto 82, Plenum Press, New York, 1983, 199–203.

[10] Chaum, D., Fiat, A., and Naor, M.*Untraceable Electronic Cash* . Springer-Verlag, 1988. 319–327

[11] Cao, T., Lin, D., and Xue, R.*A randomized RSA-based partially blind signature scheme for electronic cash.* Computers And Security, 2005, 44–49

[12] C. I. Fan, S. Y. Huang, P. H. Ho and C. L. Lei, *Fair anonymous rewarding based on electronic cash [J]* , Journal of Systems and Software, 2009, 82(7), pp. 1168-1176.

[13] Yuliang Zheng and Jennifer Seberry *Practical Approaches to Attaining Security against Adaptively Chosen Ciphertext Attacks* ,The Centre for Computer Security Research, Department of Computer Science, University of Wollongong, Locked Bag 8844 Wollongong, NSW 2521, AUSTRALIA

[14] J. Carter and M. Wegman. *Universal classes of hash functions.* Journal of Computer and System Sciences ,18:143-154,1979

[15] M. Wegman and J. Carter.*New hash functions and their use in authentication and set equality* . Journal of Computer and System Sciences, 22:265-279,1981

[16] Y. Zheng, T. Hardjono, and J. Seberry.*A practical non malleable public key cryptosystem* . Technical Report CS928 Department of Computer Science, University College, University of New South Wales,1991.

[17] Varadharajan, v., Nguyen, K. Q., and Mu, Y.*On the design of efficient rsa-based offline electronic cash schemes* . Theoretical Computer Science, 226, 1999, 173184.

[18] Liu, K., Tsang, P., and Wong, S.*Recoverable and untraceable e-cash* . In Second European PKI Workshop: Research and Applications, LNCS 3545, Springer, New York, 2005, 206–214

[19] T. ElGamal.*A public key cryptosystem and a signature scheme based on discrete logarithms.* IEEE Transactions on Information Theory, IT-31(4):469-472,1985

[20] J. Carter and M. Wegman. *Universal classes of hash functions.* Journal of Computer and System Sciences, 18:143-154,1979